

Projet Ticket To Ride

Introduction

Au cours de ce projet nous avons diverses tâches à accomplir, gérer les coups de l'adversaire (sauvegarder tous ses déplacements pour ne pas prendre des chemins déjà occupés et garder des petites informations sur ses coups pour les utiliser contre lui), gérer ses propres coups (remplir dans des structures toutes les informations nécessaires à la gestion d'une partie) et surtout trouver le chemin le plus court et l'adapter en fonction de ce que joue l'adversaire. J'ai comptabilisé 60-70 heures de programmation mais je dois avouer que je n'ai pas été particulièrement efficace à force de travailler tard le soir ou rester bloquer sur des problèmes qui ne devraient pas en être (là où il fallait juste recopier l'algorithme de Dijkstra j'ai voulu le retrouver par moi-même pour bien le comprendre et cela m'a pris au moins une quinzaine d'heures). J'ai passé autant de temps sur le projet car j'ai décidé de m'y prendre tôt car ne passant pas la deuxième semaine des vacances chez moi je n'ai pas pu vraiment travailler sur le projet, de plus j'ai vraiment essayé de faire l'effort de faire un programme commenté en anglais avec un style d'écriture qui peut s'adapter à le plus de situations possibles.

Quelques commentaires sur le code

Rédigé en anglais durant tout le programme le code contient beaucoup de commentaires pour que personne ne se sente vraiment perdu à un moment du programme (moi y compris). Un Repository github a été créé dès le début pour sauvegarder les fichiers du programme, qui contient d'ailleurs de nombreux fichiers .c et .h ainsi qu'un Makefile (avec optimisation -Ofast de gcc pour un programme encore plus rapide !). Voici le lien vers le repository qui sera passé en public le jour de la compétition :

https://github.com/PierrelouisL/TicketToRide_PL

Guide d'utilisation

Le Makefile est simple mais efficace, il suffit d'écrire make dans le terminal pour compiler tous les fichiers puis ensuite la commande « ./MainP » pour lancer le programme. Il sera demandé au début du programme de choisir un serveur à utiliser, comme indiqué il faudra taper 1 pour le serveur 1234 et n'importe quoi d'autre pour le serveur 5678.

Au lancement du code plusieurs options sont à spécifier, premièrement on indique si nous voulons aller sur le port 1234 ou 5678 puis nous informons le programme du mode utilisé, si nous voulons être en tournoi il suffit d'écrire directement le lien du tournoi, si nous voulons être en mode normal contre le NiceBot il faut écrire 00 et finalement 01 si nous voulons faire un duel.

Problèmes rencontrés

Je n'ai pas particulièrement rencontré de problème étant donné la grande marge de manœuvre donnée dans le sujet je pouvais corriger chaque bug et comprendre tout mon programme car c'était moi qui l'avais codé, cependant le dernier jour j'ai eu un petit problème du à la non mise à jour du fichier clientAPI.c mais qui a été rapidement résolu.

Conclusion

J'ai bien aimé travailler sur ce projet étant donné la satisfaction procurée par un programme qui fonctionne finalement mais ce n'a pas été le cas pour tout le monde et cela peut-être très frustrant et je sais que j'aurais été très déçu si mon programme ne fonctionnait pas. Je tenais particulièrement à vous remercier pour votre disponibilité durant toute la période de programmation. Le sujet a été bien choisi et m'a permis d'apprendre beaucoup ainsi que de consolider les connaissances que j'avais déjà.